# CLAIMS

What is claimed is:

1. A method comprising:

invoking a system management interrupt (SMI) handler in response to an SMI;

determining a thermal state of a processor by the SMI handler; and

interacting between the SMI handler and one of a speed step technology (SST) applet and a thermal driver in a thermal management operating system (OS) to transition the processor to one of a low power state and a high power state based on the thermal state according to a native performance control status.

2. The method of claim 1 wherein invoking the SMI comprises:

invoking the SMI at predetermined time intervals.

3. The method of claim 1 wherein determining the thermal state comprises:

reading a sensor indicating temperature of the processor.

4. The method of claim 1 wherein interacting comprises:

if the native performance control is enabled, interacting with the thermal management OS; and

if the native performance control is not enabled, interacting with the SST applet.

5. The method of claim 4 wherein interacting with the thermal management OS comprises:

invoking a source language code compatible with the thermal OS by the SMI handler, the source language code indicating availability status of the high power state based on the thermal state, the availability status being available if the thermal state corresponding to a low temperature and being unavailable if the thermal state corresponds to a high temperature;

exiting the SMI handler;

invoking the source language code by the thermal driver;

executing the source language code, the executed source language code notifying a processor object of the availability status of the high power state via a present performance capability structure; and

transitioning the processor to the low power state if the availability status is unavailable and to one of a current state and the high power state if the availability status is available.

6. The method of claim 5 wherein interacting comprises:

interacting between the SMI handler and the thermal driver in an advanced configuration and power management (ACPI) operating system (OS).

7. The method of claim 6 wherein invoking the source language code comprises:

invoking an ACPI source language code (ASL).

8. The method of claim 4 wherein interacting with the SST applet comprises:

transitioning the processor to a last requested performance state in the SST applet if the thermal state corresponds to a low temperature;

saving current processor performance state in the SST applet if the thermal state corresponds to a high temperature;

transitioning the processor to the low power state if the thermal state corresponds to a high temperature; and

exiting the SMI handler.

9. The method of claim 1 further comprising:

processing an SST command using the SST applet.

10. The method of claim 1 wherein processing the SST command comprises:

returning a current processor state if the SST command is a get status command;

recording a requested stated if the SST command is a set state command and the thermal state corresponds to a high temperature; and

transitioning the processor to a last requested state and recording the current processor state if the SST command is a set state command and the thermal state corresponds to a low temperature.

11. A computer program product comprises:

a machine useable medium having computer program code embedded therein, the computer program product having:

computer readable program code to invoke a system management interrupt (SMI) handler in response to an SMI;

computer readable program code to determine a thermal state of a processor by the SMI handler; and

computer readable program code to interact between the SMI handler and one of a speed step technology (SST) applet and a thermal driver in a thermal management operating system (OS) to transition the processor to one of a low power state and a high power state based on the thermal state according to a native performance control status.

12. The computer program product of claim 11 wherein the computer readable program code to invoke the SMI comprises:

computer readable program code to invoke the SMI at predetermined time intervals.

13. The computer program product of claim 11 wherein the computer readable program code to determine the thermal state comprises:

computer readable program code to read a sensor indicating temperature of the processor.

14. The computer program product of claim 11 wherein the computer readable program code to interact comprises:

computer readable program code to interact with the thermal management OS if the native performance control is enabled; and

computer readable program code to interact with the SST applet if the native performance control is not enabled.

15. The computer program product of claim 14 wherein the computer readable program code to interact with the thermal management OS comprises:

computer readable program code to invoke a source language code compatible with the thermal OS by the SMI handler, the source language code indicating availability status of the high power state based on the thermal state, the availability

status being available if the thermal state corresponding to a low temperature and being unavailable if the thermal state corresponds to a high temperature;

computer readable program code to exit the SMI handler;

computer readable program code to invoke the source language code by the thermal driver;

computer readable program code to execute the source language code, the executed source language code notifying a processor object of the availability status of the high power state via a present performance capability structure; and

computer readable program code to transition the processor to the low power state if the availability status is unavailable and to one of a current state and the high power state if the availability status is available.

16. The computer program product of claim 15 wherein the computer readable program code to interact comprises:

computer readable program code to interact between the SMI handler and the thermal driver in an advanced configuration and power management (ACPI) operating system (OS).

17. The computer program product of claim 16 wherein the computer readable program code to invoke the source language code comprises:

computer readable program code to invoke an ACPI source language code (ASL).

18. The computer program product of claim 14 wherein the computer readable program code to interact with the SST applet comprises:

computer readable program code to transition the processor to a last requested performance state in the SST applet if the thermal state corresponds to a low temperature;

computer readable program code to save current processor performance state in the SST applet if the thermal state corresponds to a high temperature;

computer readable program code to transition the processor to the low power state if the thermal state corresponds to a high temperature; and

computer readable program code to exit the SMI handler.

19. The computer program product of claim 11 further comprising:

computer readable program code to process an SST command using the SST applet.

20. The computer program product of claim 11 wherein the computer readable program code to process the SST command comprises:

computer readable program code to return a current processor state if the SST command is a get status command;

computer readable program code to record a requested stated if the SST command is a set state command and the thermal state corresponds to a high temperature; and

computer readable program code to transition the processor to a last requested state and recording the current processor state if the SST command is a set state command and the thermal state corresponds to a low temperature.

21. A system comprising:

a processor;

a memory coupled to the processor to store a thermal management module, the thermal management module including a system management interrupt (SMI) handler and a thermal management operating system (OS), the thermal management module, when executed, causing the processor to:

invoke a system management interrupt (SMI) handler in response to an SMI,

determine a thermal state of a processor by the SMI handler, and

interact between the SMI handler and one of a speed step technology (SST) applet and a thermal driver in a thermal management operating system (OS) to transition the processor to one of a low power state and a high power state based on the thermal state according to a native performance control status.

22. The system of claim 21 wherein the thermal management module causing the processor to invoke the SMI causes the processor to:

invoke the SMI at predetermined time intervals.

23. The system of claim 21 wherein the thermal management module causing the processor to determine the thermal state causes the processor to:

read a sensor indicating temperature of the processor.

24. The system of claim 21 wherein the thermal management module causing the processor to interact causes the processor to:

interact with the thermal management OS if the native performance control is enabled; and

interact with the SST applet if the native performance control is not enabled.

25. The system of claim 24 wherein the thermal management module causing the processor to interact with the thermal management OS causes the processor to:

invoke a source language code compatible with the thermal OS by the SMI handler, the source language code indicating availability status of the high power state based on the thermal state, the availability status being available if the thermal state corresponding to a low temperature and being unavailable if the thermal state corresponds to a high temperature;

exit the SMI handler;

invoke the source language code by the thermal driver;

execute the source language code, the executed source language code notifying a processor object of the availability status of the high power state via a present performance capability structure; and

transition the processor to the low power state if the availability status is unavailable and to one of a current state and the high power state if the availability status is available.

26. The system of claim 25 wherein the thermal management module causing the processor to interact causes the processor to:

interact between the SMI handler and the thermal driver in an advanced configuration and power management (ACPI) operating system (OS).

27. The system of claim 26 wherein the thermal management module causing the processor to invoke the source language code causes the processor to:

invoke an ACPI source language code (ASL).

28. The system of claim 24 wherein the thermal management module causing the processor to interact with the SST applet causes the processor to:

transition the processor to a last requested performance state in the SST applet if the thermal state corresponds to a low temperature;

save current processor performance state in the SST applet if the thermal state corresponds to a high temperature;

transition the processor to the low power state if the thermal state corresponds to a high temperature; and

exit the SMI handler.

29. The system of claim 21 the thermal management module, when executed, further causes the processor to:

process an SST command using the SST applet.

30. The system of claim 21 wherein the thermal management module causing the processor to process the SST command causes the processor to:

return a current processor state if the SST command is a get status command;

record a requested stated if the SST command is a set state command and the thermal state corresponds to a high temperature; and

transition the processor to a last requested state and recording the current processor state if the SST command is a set state command and the thermal state corresponds to a low temperature.